Sandia
National
Laboratories

# Xyce™ XDM Netlist Translator User Guide, Version 2.1

Gary J. Templet, Garrick Ng, Richard L. Schiek, Peter E. Sholander, Jason C. Verley

## ABSTRACT

This manual describes the installation and use of the **Xyce**™ XDM Netlist Translator. XDM simplifies the translation of netlists generated by commercial circuit simulator tools into Xyce-compatible netlists. XDM currently supports translation from PSpice and HSPICE netlists into **Xyce**™ netlists.

## ACKNOWLEDGMENTS

### *Trademarks*

Xyce Electronic Simulator™ and **Xyce**™ are trademarks of National Technology & Engineering Solutions of Sandia, LLC (NTESS).

### *Contact Information*

Address

Electrical Models & Simulation Dept.
Sandia National Laboratories
P.O. Box 5800, MS 1177
Albuquerque, NM 87185-1177

**Outside Sandia**

| | |
|---|---|
| World Wide Web | http://xyce.sandia.gov |
| Email | xyce@sandia.gov |

**Inside Sandia**

| | |
|---|---|
| World Wide Web | http://xyce.sandia.gov |
| Email | xyce-sandia@sandia.gov |
| Bug Reports | http://joseki-vm.sandia.gov/bugzilla |
| | http://morannon.sandia.gov/bugzilla |

# CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

# 1.   INTRODUCTION

The Xyce XDM Netlist Translator is a software solution developed by Sandia National Laboratories for translating SPICE netlists into a format readable by Xyce, Sandia's internally developed SPICE-compatible simulation tool.

XDM can currently read netlists in PSpice and HSPICE formats. The PSpice translation capabilities are the most mature, and can be considered at a production quality. The HSPICE translation was new in version 2.0 and has not been extensively tested "in the wild," so we welcome feedback from users. Xyce has an option, `-hspice-ext`, to handle HSPICE syntax not fully translated by XDM.

XDM can be built on a number of platforms including Linux, OS X and Windows. Sandia supplies binaries for RHEL7, OS X and Windows.

## 2.      XDM INSTALLATION

This chapter describes how to install and use XDM. The procedure is substantially the same on Windows, Linux, and OS X PCs.

### 2.1.      Windows

The XDM binary is supplied as a "zip" file. Open the zipped file by double clicking it. Copy the unzipped folder to a location of your choice; we recommend the same directory in which Xyce was installed. By default, the Xyce package is installed in the `C:\Program Files\` folder. In the following, it is assumed that the XDM folder is called `xdm-2.0.0-win64`.

Inside the `xdm-2.0.0-win64\bin` folder, a batch file, called `xdm_console`, is provided to automatically put the XDM binary in the path (so that the full path does not need to be invoked each time XDM is run). For convenience, create a link to the `xdm_console` file on your desktop by right clicking on the file and selecting "Create shortcut."

### 2.2.      Linux and OS X

The XDM binary is supplied as a "zip" file. Simply unzip it in the location from which you want to run it. It is recommended to place binary in the same directory in which Xyce was installed; on Linux and OS X systems, the default location is the `/usr/local` directory.

Once the XDM directory is in the preferred location, add the installed location of XDM to the PATH variable, so you don't have to type the whole path to the binary every time. Do so by editing the shell start-up file (e.g., `.bashrc` if your shell is bash, or `.cshrc` if the shell is the c-shell). Exact syntax depends on the shell used, but for bash and zsh the syntax is (using OS X as the example):

```
export PATH=$PATH:/usr/local/xdm-2.0.0-Darwin/bin
```

The exact path will depend on the installed version of XDM. Look in `/usr/local` with the following command:

```
ls -l /usr/local
```

to identify the actual install directory used by the installed version of XDM. Once entered into the start-up file, the path will be set this way at the next log in. The same command can be issued directly in the command line and it will take effect immediately.

# 3.   TRANSLATING NETLISTS WITH XDM

## 3.1.   Startup

### 3.1.1.   Windows

If you created a link to the `xdm_console` batch file (as described in Chapter 2), then double-click the file. Otherwise, start the "Command Prompt". The remaining instructions assume the XDM binary is in your path.

### 3.1.2.   Linux/OS X

Start a terminal shell. It is assumed that you have the XDM binary in your path.

## 3.2.   Running XDM

An example XDM invocation for translating a file is

```
xdm_bdl -s hspice -d out -o xyce --auto test_circuit.sp
```

where

- `-s hspice` designates an HSPICE input format

- `-d out` designates writing output to a directory called, "out"

- `-o xyce` designates the Xyce output format

- `--auto` specifies that XDM should automatically translate any `.INC`/`.LIB` files

- `test_circuit.sp` is the input circuit to be translated

To see a full description of XDM's flags, type `xdm_bdl -h` on the command line:

```
$ xdm_bdl -h
usage: xdm_bdl [-h] [-s [{hspice,tspice,pspice,spectre,xyce}]] [-d [DIR_OUT]]
               [-o [{xyce}]] [--auto] [--eval] [-l {DEBUG,INFO,WARN,ERROR}]
               [-q {R,C,D,L,X,Q,ALL}] [--license]
               input_file

xdm 2.0.0: Translates input netlist file by creating a new netlist file of a
different netlist file format. The translated input file (of the same name) is
written into the specified output directory - if you used the same directory as
the input netlist file, the original file will be overwritten. xdm also
```

```
supports a device query interface for the SAW environment.

positional arguments:
  input_file              The input netlist file

optional arguments:
  -h, --help              show this help message and exit
  -s [{hspice,tspice,pspice,spectre,xyce}],
          --source_file_format [{hspice,tspice,pspice,spectre,xyce}]
                          The source/input netlist file format (default: pspice)
  -d [DIR_OUT], --dir_out [DIR_OUT]
                          The output directory (default: default_dir)
  -o [{xyce}], --output_file_format [{xyce}]
                          The output netlist file format (default: xyce)
  --auto                  Automatically translate include and library files
                          (default: False)
  --eval                  Evaluate functions during translation (default: False)
  -l {DEBUG,INFO,WARN,ERROR}, --logging {DEBUG,INFO,WARN,ERROR}
                          Control the level of screen logging output: WARN is
                          quiet - only ERROR and WARN level messages will be
                          sent to the screen (default: WARN)
  -q {R,C,D,L,X,Q,ALL}, --query_device {R,C,D,L,X,Q,ALL}
                          Query for a device type of interest within the SAW
                          environment (default: None)
  --license               Display the license for this version of XDM (default:
                          None)
```

Note that, while several input netlist formats are listed, only HSPICE and PSpice have functional support.

After XDM runs, the translated circuit file and its associated library files in the above example should be in the "out" directory. To run the circuit file in Xyce, the command:

```
Xyce test_circuit.sp -hspice-ext all
```

should be used to resolve any remaining translation issues not handled by XDM.

When XDM encounters something it doesn't understand, it will either produce a warning message, and continue processing, or print an error message and stop processing. Typically, a warning message is produced when a specific line cannot be translated. In these cases, the user can choose to ignore the issue, or may be able to find an equivalent Xyce syntax by hand. An example warning message is shown in Figure 3-1. The line, in the input file, that caused the warning message is then typically left as a comment line in the translated netlist.

Error messages occur in cases where XDM cannot continue processing, such as a missing file, or non-ASCII characters appearing in the netlist. XDM will give as much information as possible for the source of the error. An example warning message is shown in Figure 3-2.

Text of commented-out PSpice line

```
xdm_bdl.exe 1.4.0 (last changed on 2017-08-04 14:46:44) at
            C:\Users\Desktop\xdm-1.4.0-win64\bin\xdm_bdl

is translating the file:

            'pwl_sources.cir.pspice' (input format = pspice)
                    using xml definition C:\Users\AppData\Local\Temp\_MEI13~2\pspice.xml
            => and is creating the translated files under the directory 'pwl_sources.cir-translated' (output format = xyce)
                    using xml definition C:\Users\AppData\Local\Temp\_MEI13~2\xyce.xml

Original calling command for this run was:

    xdm_bdl.exe '-s' 'pspice' '-d' 'pwl_sources.cir-translated' '-o' 'xyce' 'pwl_sources.cir.pspice'


            08/07/2017 10:00:55 AM WARNING:  In file:"SCHEMATIC1.net" at line:[32, 33]. * V_V5 1 0 PWL REPEAT FOREVER
(0,0) (5e-4,1) (1e-3,0)  ENDREPEAT; PSpice Parser Retained (as a comment). Continuing.

=== xdm execution complete:

    Total critical issues reported                          = 0:
    Total       errors reported                             = 0:
    Total       warnings reported                           = 1:
    Total       information messages reported               = 5:

SUCCESS: xdm completion status flag = 0:
```

"0" = success

**Figure 3-1. Example of an XDM Warning message.**

```
xdm_bdl.exe 1.4.0 (last changed on 2017-08-04 14:46:44) at
            C:\Users\Desktop\xdm-1.4.0-win64\bin\xdm_bdl

is translating the file:

    'example1.cir' (input format = pspice)
            using xml definition C:\Users\AppData\Local\Temp\_MEI13~2\pspice.xml
    => and is creating the translated files under the directory 'example1.cir-translated' (output format = xyce)
            using xml definition C:\Users\AppData\Local\Temp\_MEI13~2\xyce.xml

Original calling command for this run was:

    xdm_bdl.exe '-s' 'pspice' '-d' 'example1.cir-translated' '-o' 'xyce' 'example1.cir'


    08/07/2017 01:09:02 PM ERROR:  File: C:\XDM_1.4.0_Windows_Test\Bogo.lib was not found. Please locate this file and try again.
```

PSpice file name

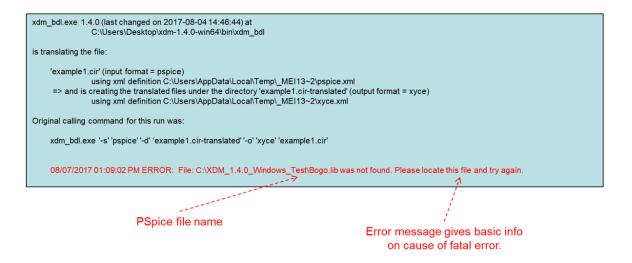Error message gives basic info
on cause of fatal error.

**Figure 3-2. Example of an XDM Error message.**

15

# 4. HSPICE TRANSLATIONS: OVERVIEW AND KNOWN ISSUES

**HSPICE `.OPTIONS`**   With one exception, HSPICE `.OPTIONS` lines will be commented out in the translated Xyce netlist for the following reasons:

- There may not be a direct translation of HSPICE options into Xyce options.

- Even options with the same or similar names in HSPICE and Xyce may not have the same functionality.

- The exception is the TNOM option in HSPICE, which will be translated in the Xyce netlist.

**Output directory**   XDM will output all translated files to the same directory, either to the default `out` directory, or to the user specified directory through the command line option, `-d [output_directory]`. However, the path to the library files in `.LIB` will remain in place. If the user wishes to run the simulation in the output directory, the path in the file name needs to be removed.

**Instance parameters**   Issues to keep in mind with regards to translations of instance parameters:

- Instance parameters of devices that exist in HSPICE but not in Xyce will be automatically removed. E.g., `DTEMP` will be removed for for R, L and C devices. *Watch for this in the on-screen warnings.*

- Care needs to be used when deciding whether to translate with the `-auto` option. If a device is an instantiation of a model declared in another file, and auto translate is turned off, that device's instance parameters will default to parameters using a default level (typically `level=1`). If the different model levels have different allowed instance parameters, this may result in parameters being removed.

**Wildcards**   Wildcards in `.PRINT` statements are generally illegal in Xyce, and will be commented out of the resulting netlist by XDM. The only exception is `V(*)`, which is allowed and will be translated by XDM.

**Multiplicity (M Factor)**   In HSPICE, the "multiplicity" (or "M Factor") can be used to specify multiple netlist devices in parallel via a single instance line. In Xyce, the terms "multiplicity factor" and "multiplier" are used to describe that same concept.

At present, the multiplicity factor (`M` parameter) is only supported in Xyce by the R, L, C and MOSFET device models, and some BJT device models (VBIC 1.3 and MEXTRAM). It is not supported for the X device (subcircuits).

17

**Solution-Dependent Resistors**   In HSPICE, it is legal to have the resistance value for an R device depend on a solution variable. An example is as follows, where `P1` is a parameter defined elsewhere in the netlist:

```
R1   1   2   'P1*v(in,out)'
```

Xyce does not support solution-dependent resistors, but the equivalent behavior can be specified using the Xyce Nonlinear Dependent Source (B device). XDM will automatically make this translation. However, it is important to note that, if the resistor value becomes zero, the expression becomes undefined, and overflows will occur. For example, the above resistor would be translated to the following:

```
B1   1   2   I={(V(2)-V(1)/(P1*v(in,out))}'
```

If, during the simulation the pin voltages are the same (`in=out`), then `v(in,out)=0`, and the expression results in an infinite current.

**AGAUSS and AUNIF**   The `AGAUSS` and `GAUSS` functions are defined both in HSPICE and Xyce to handle Gaussian distributions. For uniform distributions, HSPICE then uses the `AUNIF` and `UNIF` functions, while Xyce uses the `RAND` function. The Xyce definitions are given in the "Expressions" section of the Xyce Reference Guide [1]. The HSPICE and Xyce versions of `AGAUSS`, `GAUSS`, and their respective `AUNIF`, `UNIF` and `RAND` functions, are not yet fully compatible:

- In HSPICE, if Monte Carlo (MC) sampling is *not* turned on, then the `AGAUSS` and `GAUSS` functions just return the mean of the distribution. If MC sampling is turned on, then HSPICE will randomly sample the specified distributions.

- The default Xyce behavior for the `AGAUSS` and `GAUSS` functions is to return a random number from the specified Gaussian distribution. The "`-hspice-ext random`" command line option will make them return the mean (see Section 4.2).

## 4.1.    Commands known not to translate

The following command lines, found in HSPICE, are not directly supported in Xyce:

- .ALTER

- .TEMP

- .IF, .ELSEIF, .ELSE and .ENDIF

In addition, any usages of `DTEMP` for subcircuits will likely need to be replaced with explicit parameter values, especially for `DTEMP` not equal to 0.

**.NODESET**   The Xyce `.NODESET` command uses a different strategy than either SPICE3F5 or HSPICE. So, the Xyce behavior may differ from that provided by `.NODESET` and `.OPTION DCHOLD` in HSPICE. In addition, Xyce does not allow the use of "wildcards" in `.NODESET` (or `.IC`) statements. The ".*NODESET (Approximate Initial Condition, Bias point)*" section of the Xyce Reference Guide [1] gives more details on the Xyce implementation.

**.OPTION MACMOD**   The various HSPICE MACMOD options allow HSPICE to search for subcircuit definitions or Verilog-A definitions in place of model references and/or vice-versa (depending on the specification). Xyce does not support this option at this time.

**Verilog-A Support**   Xyce does have the capability to dynamically link in Verilog-A models. However, that capability is limited and not HSPICE compatible. In particular, it is not possible to insert Verilog-A models into Xyce via the netlist alone. So, Xyce does not support the HSPICE `.HDL` command.

**Multiple .END Statements**   The following netlist is legal in HSPICE.

```
Multiple .END statements
*********************************
V1 1 0 SIN(0 1 1e3)
R1 1 2 1
R2 2 0 2
.TRAN 10u 1m
.PRINT TRAN V(1) V(2)
.END

V1 1 0 SIN(0 1 1e3)
R1 1 2 1
R2 2 0 3
.TRAN 10u 1m
.PRINT TRAN V(1) V(2)
.END
```

Both simulations will be run, once with the resistance of R2 equal to 2 and once with its resistance equal to 3. In Xyce, the simulation would only be run with the resistance of R2 equal to 2. All of the text after the first `.END` statement would be treated as comment lines by the Xyce parser. To run both simulations in Xyce, the appropriate `.STEP` or `.DATA` statement would be used to set the desired values for the resistance of R2.

## 4.2.     Xyce `-hspice-ext` Command Line Option

The Xyce command line option, `-hspice-ext`, is almost always required when running a netlist translated by XDM. It allows the Xyce parser to accept a limited set of HSPICE syntax features, in lieu of the Xyce ones, for the limited set of cases shown in Table 4-1. These features are particularly difficult for XDM to translate, but are easy for Xyce to handle if it knows about them.

**Table 4-1. `-hspice-ext` Command Line Option**

| Syntax | Functionality |
|---|---|
| `-hspice-ext units` | Toggles on `A=1e-18` as a scaling factor |

**Table 4-1. `-hspice-ext` Command Line Option**

| Syntax | Functionality |
| --- | --- |
| `-hspice-ext math` | Toggles in the HSPICE meanings for the logical operators `||` and `&&` and the exponentiation synonym of `^`. It also uses the HSPICE functions for base10 and natural logarithms, which are `LOG10` and `LOG`. |
| `-hspice-ext random` | `AGAUSS()` and `GAUSS()` will return the mean value rather than a random variate |
| `-hspice-ext all` | Does all three |
| `-hspice-ext units,math,random` | A comma-separated listing is also legal. This example is equivalent to `all`. |

## 5.        TRANSLATION OF HSPICE-BASED PDK MODEL LIBRARIES

The entirety of an HSPICE-based PDK model library can be converted by translating a circuit that includes the model library through the `.INC` or the `.LIB` file and using the `--auto` option.

To translate the PDK without an accompanying circuit, create a dummy file that has just single include statement for the top level wrapper file, e.g.:

```
.INC design.inc
or
.LIB design_wrapper.lib TT
```

and translate that with the `--auto` option.

It is recommended to translate a PDK's model library all at once using the `--auto` option, since XDM does some cross-checking of models and may remove parameters and/or comment out lines if all the information in a PDK is processed at the same time (see Chapter 4 for more details).

In the current release of Xyce, it may be necessary to pre-evaluate certain functions in the model library using XDM in order for a circuit to run using Xyce. If a Xyce simulation of a circuit hangs and does not run to completion, users may try to translate the circuit with the pre-evaluation flag on. This is done using the `--eval` option in addition to the `--auto` option at the command line.

- In this mode, XDM will evaluate functions in the model library based on the .PARAM statements and device instance parameters listed in the circuit, and will write the values of those evaluations into the circuit and model library files.

- The resulting translated model library will be specific to the circuit, and *cannot* be used in general with other circuits designed with the same model library. Users will have to re-run the translation again on different circuits to get the translated model library specific to that circuit.

- The function evaluation and pre-processing may take up to several minutes to complete, especially if the circuit includes functions that involve a high degree of nesting.

# 6. PSPICE TRANSLATIONS: OVERVIEW AND KNOWN ISSUES

**Libraries and Models**   XDM does not throw a fatal error or emit a warning if a `.MODEL` statement is missing for a model that requires one. As a result, Xyce will emit an error message during netlist parsing.

**Unsupported Device Groups**   XDM does not yet support the following groups of devices, because of differences in syntax and parameters:

- Digital device models

- T devices (lossy/lossless transmission lines)

**Unsupported PSpice Syntax**   Some PSpice options and commands do not have a legal Xyce translation. The following are always commented out by XDM, but should not affect the Xyce simulation:

- `.AUTOCONVERGE`

- `.OPTIONS ADVCONV`

- `.OPTIONS CHGTOL`

- `.OPTIONS ITL2`

- `.OPTIONS VNTOL`

**Undocumented PSpice Syntax**   Some PSpice syntax is undocumented in the PSpice guides. The following are known examples (for device instance lines and `.MODEL` statements) that aren't supported by XDM:

- Missing commas in TC specifications
  `TC=0.1 0.1`
  vs.
  `TC=0.1,0.1`

    – The latter is what is documented. The former will be commented out by XDM.

- Missing parameter values
  `TC=` or `BV=` without a value

- Extra matched set of parentheses
  `.MODEL D1N3940 D((BV=600 DEV=1) IS=4E-10 RS=.105 N=1.48 CJO=1.95E-11)`

- Unmatched right/left parentheses in `.MODEL` statements
  `.MODEL D1N3940 D((BV=600 DEV=1%) IS=4E-10 RS=.105 N=1.48 CJO=1.95E-11)`

**Piecewise Linear (PWL) Sources**   XDM is able to handle these PSpice forms of PWL:

- Documented syntax in the PSpice Reference Guide
  ```
  V_V1 1 0 PWL (1e-3,0.5)(2e-3,1)(3e-3,1)(4e-3,0.5)
  ```

- Undocumented syntax in the PSpice Reference Guide
  ```
  V_V1 1 0 PWL 1e-3 0.5 2e-3 1 3e-3 1 4e-3 0.5
  V_V2 2 0 PWL (1e-3, 0.5, 2e-3, 1, 3e-3, 1, 4e-3, 0.5)
  V_V3 3 0 PWL (1e-3 0.5 2e-3 1 3e-3 1 4e-3 0.5)
  ```

- Using a file
  ```
  V_FILE1 5 0 PWL FILE "pwlFile1.txt"
  ```

The following PSpice forms of PWL will be commented out by XDM, since they don't have a legal (or straightforward) translation in Xyce:

- Some instances of REPEAT
  See Section 6.1.12, "*Piecewise Linear Sources*" of the Xyce Reference Guide [1] for more details on how to manually translate these PSpice PWL instance lines.

- Anything using TIME_SCALE_FACTOR or VALUE_SCALE_FACTOR

**Controlled sources**   While the documented syntax of the POLY form works for the F and G sources, the subcircuit syntax produced by OrCAD capture is not always consistent with the documentation.

**.PROBE and .PROBE64**   Some of the wildcard forms used by .PROBE and .PROBE64 are not supported:

- /V( ) all voltages

- -/V( ) no voltages

- -/V(X) all voltages except internal sub-circuit voltages

## REFERENCES

[1] Eric R. Keiter, Thomas V. Russo, Richard L. Schiek, Heidi K. Thornquist, Ting Mei, Jason C. Verley, Peter E. Sholander, and Karthik V. Aadithya. Xyce Parallel Electronic Simulator: Reference Guide, Version 7.0. Technical Report SAND2020-1214 O, Sandia National Laboratories, Albuquerque, NM, 2020.

## APPENDIX A.  Third Party Licenses

The Xyce XDM Netlist Translator makes use of code developed by various third parties. The following text is provided to comply with the licenses of the codes that require it.

XDM uses Boost:

```
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or
organization obtaining a copy of the software and accompanying
documentation covered by this license (the "Software") to use,
reproduce, display, distribute, execute, and transmit the Software, and
to prepare derivative works of the Software, and to permit third-parties
to whom the Software is furnished to do so, all subject to the
following:

The copyright notices in the Software and this entire statement,
including the above license grant, this restriction and the following
disclaimer, must be included in all copies of the Software, in whole or
in part, and all derivative works of the Software, unless such copies or
derivative works are solely in the form of machine-executable object
code generated by a source language processor.
```

XDM uses the Python 2.7 Standard Library:

```
Python is Copyright 2001-2020 Python Software Foundation; All Rights
Reserved

1. This LICENSE AGREEMENT is between the Python Software Foundation
   ("PSF"), and the Individual or Organization ("Licensee") accessing
   and otherwise using Python 2.7.17 software in source or binary form
   and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF
   hereby grants Licensee a nonexclusive, royalty-free, world-wide
   license to reproduce, analyze, test, perform and/or display publicly,
   prepare derivative works, distribute, and otherwise use Python 2.7.17
   alone or in any derivative version, provided, however, that PSF's
   License Agreement and PSF's notice of copyright, i.e., "Copyright ©
   2001-2020 Python Software Foundation; All Rights Reserved" are
   retained in Python 2.7.17 alone or in any derivative version prepared
   by Licensee.
```

3. In the event Licensee prepares a derivative work that is based on or incorporates Python 2.7.17 or any part thereof, and wants to make the derivative work available to others as provided herein, then Licensee hereby agrees to include in any such work a brief summary of the changes made to Python 2.7.17.

4. PSF is making Python 2.7.17 available to Licensee on an "AS IS" basis.  PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED.  BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON 2.7.17 WILL NOT INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON 2.7.17 FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON 2.7.17, OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any relationship of agency, partnership, or joint venture between PSF and Licensee.  This License Agreement does not grant permission to use PSF trademarks or trade name in a trademark sense to endorse or promote products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python 2.7.17, Licensee agrees to be bound by the terms and conditions of this License Agreement.

XDM uses the RapidXML Library:

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## DISTRIBUTION

**Hardcopy—Internal**

| Number of Copies | Name | Org. | Mailstop |
|---|---|---|---|
| 1 | Technical Library | 9536 | 0899 |